CLAIMS

1.   A method for instrumenting a software program, the method comprising:

embedding within the software program a user-defined measurement type instance based one of a plurality of predefined measurement types; and

storing data structures representing the embedded user-defined measurement type instance.

2.   The method of claim 1 wherein the user-defined measurement type instance includes a measurement type designation and associated variables.

3.   The method of claim 2 wherein the user-defined measurement type instance is associated with a data state comprising the current values of the associated variables, and wherein the data structures representing the embedded user-defined measurement type instance are created and maintained by routines of a library.

4.   The method of claim 3 wherein the predefined measurement types include:

an atomic measurement type, a data set for which can be transmitted to the library at points in time selected by the software program;

a transaction measurement type, data collection for which can be started and ended by the software program, and, upon ending, results in transmission of a data set to the library; and

a polling measurement type, data sets for which are obtained by the library from the software program at regular intervals.

5.   The method of claim 3 wherein the additional data structures representing aggregate measurement type instances based on the embedded user-defined measurement type instance are generated and stored by the library, the aggregate

measurement types accumulated data over a period of time from data states generated for the embedded user-defined measurement type instance.

6.   The method of claim 5 wherein aggregate measurement types include:

a count measurement type;

a group measurement type;

a sum measurement type; and

a threshold measurement type.

7.   A method for collecting data by a library from a user-defined measurement type instance embedded within a software program, the user-defined measurement type instance having a measurement type selected from among an atomic measurement type, a transaction measurement type, and a polling measurement type, and the library containing one or more additional aggregation measurement types based on the user-defined measurement type, the method comprising:

receiving data states, comprising current values of variables associated with the embedded user-defined measurement type instance, generated during execution of the software program;

filtering the data states;

processing data contained in the data states to produce output data according to one or more measurement type instances with which the data states are associated; and

packaging output data into reports.

8.   The method of claim 7 wherein filtering of a data state further includes applying a filter expression to the variable values contained in the data state to determine whether or not to accept the data state for further processing.

9.   The method of claim 8 wherein a filter expression may be associated with each measurement type instance.

10. The method of claim 7 wherein processing data contained in a data state to produce output data further includes:

selecting each measurement type instance associated with the data state;

for each selected measurement type,

if the measurement type is derived from the atomic or polling measurement type, including the variable values in the data state in the output data,

if the measurement type is derived from the transaction data type, including the variable values in the in the data state in the output data along with a calculated value equal to a transaction duration; and

if the measurement type is an aggregate measurement type, accumulating data for subsequent data output.

11. The method of claim 7 further including transmitting the reports to a data analysis component.

12. The method of claim 7 further including transmitting a description of the embedded user-defined measurement type instance and any aggregate measurement type instances based on the embedded user-defined measurement type instance to a data analysis component, which then transmits configuration information back to the library to control collection and reporting of output data.

13. A software program instrumentation system comprising:

a library that manages a data structure that represents a software program with an embedded user-defined measurement type instance having a measurement type selected from among predefined measurement types and associated with variables, the library managing data structures that represent the embedded user-defined measurement type, the embedded user-defined measurement type instance, and one or more additional aggregate measurement types based on the embedded user-defined measurement type instance, the library collecting data states generated from current values of the variables associated with the embedded user-defined measurement type instance and generating reports based on the collected data states; and

a data analysis routine that configures the data structures by altering values of fields within the data structures to control data collection and reporting by the library, and that receives the generated reports from the library.

14. The software program instrumentation system of claim 13 wherein the predefined measurement types include:

an atomic measurement type, a data set for which can be transmitted to the library at points in time selected by the software program;

a transaction measurement type, data collection for which can be started and ended by the software program, and, upon ending, results in transmission of a data set to the library; and

a polling measurement type, data sets for which are obtained by the library from the software program at regular intervals.

15. The software program instrumentation system of claim 13 wherein the one or more additional aggregate measurement types are derived from predefined aggregate measurement types including:

a count measurement type that accumulates, over a data collection interval, a count of data states associated with the embedded user-defined measurement type instance;

a group measurement type that accumulates, over a data collection interval, a count of data states associated with the embedded user-defined measurement type instance and some number of additional embedded user-defined measurement type instances;

a sum measurement type that accumulates, over a data collection interval, statistical quantities derived from a variable value within the data states; and

a threshold measurement type that accumulates, over a data collection interval, a count of data states associated with the embedded user-defined measurement type instance that contain variable values that produce an acceptance output from a threshold test expression.

16. The software program instrumentation system of claim 15 wherein the sum measurement type accumulates components of statistical quantities, including median,

average, and standard deviation, so the running medians, averages, and standard deviations may computed over time from multiple reports.

17. The software program instrumentation system of claim 13 wherein the library collects data states generated from current values of the variables associated with the embedded user-defined measurement type instance by:

filtering the data states based on filtering criteria associated with each measurement type instance in order to direct a data state to each measurement type instance with which it is associated providing that the data state is accepted for the measurement type instance by the filtering criteria;

for an accepted data state directed to a transaction measurement type instance, calculating a duration value for the transaction and including that duration value in the data state;

for an accepted data state directed to an aggregation measurement type instance, processing the data state according to the aggregation measurement type of the aggregation measurement type instance; and

outputting the data state for the user-defined measurement type instance, when the user-defined measurement type instance is configured for output, into a report; and

outputting accumulated data for aggregation measurement type instances at a next accumulation interval into a report.

18. The software program instrumentation system of claim 13 wherein an aggregate measurement type instance may include key variables that control partitioning of accumulated data into a set of accumulated data, each accumulated data of the set output into a report at a next accumulation interval by the library.

19. The software program instrumentation system of claim 18 wherein only the highest values within a set of accumulated data are output into a report at a next accumulation interval by the library.

20.   The software program instrumentation system of claim 13 wherein the reports are XML: documents.